ANTHROP\C

Activating AI Safety Level 3 Protections

May 2025

Abstract

In this report, we present Anthropic's implementation of AI Safety Level 3 (ASL-3) Deployment and Security Standards. The ASL-3 Standard, set forth in Anthropic's Responsible Scaling Policy (RSP), aims to prevent misuse of our models by making them harder to use for catastrophic harm and by protecting model weights from being illicitly obtained. Although our most recent model (Claude Sonnet 3.7) did not require ASL-3 protections, we decided to take steps to implement the ASL-3 Standard in anticipation of near future model releases which we believe face an increasing risk of meeting this threshold.

We are now releasing Claude Opus 4 as our first deployment of an AI model under the heightened protections of ASL-3. Although we have not determined whether the ASL-3 Standard is required for this model, Claude Opus 4's clearly superior performance on key RSP evaluations as compared to Claude Sonnet 3.7 meant that we could not rule out the need for the ASL-3 Standard, and we are implementing it now as a precautionary measure. We recognize that our initial implementation will almost certainly not be perfect, and we hope to rapidly learn, iterate, and debug it.

The report details the components of our approach to the ASL-3 Standard, which fall into two categories: (1) deployment safeguards focused on addressing universal jailbreaks relevant to our key threat models; and (2) security measures protecting model weights against non-state threat actors. Key deployment measures include classifier-based guards that monitor model inputs and outputs and intervene to block a narrow class of harmful information. Security enhancements include egress bandwidth controls, change management protocols, endpoint software control, and two-party authorization systems. Safety and security are evolving challenges requiring persistent attention and adaptation. As we learn more, we may decide to change or iterate on the measures described here; we may also turn off these measures if a model previously under ASL-3 protections is determined not to cross the ASL-3 capabilities threshold. In the meantime, we are sharing our approach to implementing the ASL-3 Standards as part of the industry- and society-wide discourse that will facilitate safe and secure development of AI.

Contents

1 Introduction

This report introduces Anthropic's approach to achieving the AI Safety Level 3 (ASL-3) Deployment and Security Standards. In this section we outline our high-level approach to security and safety and some considerations related to the timing of meeting these standards.

The initial deployment of models under the ASL-3 Standard is part of an ongoing process. Our deployment and security protections can be improved, and we are committed to learning from our operational experience to enhance safety measures and strengthen our security protocols.

1.1 Security and deployment standards

Anthropic's Responsible Scaling Policy (RSP) sets forth our commitment not to train or deploy models capable of causing catastrophic harm unless we have implemented safety and security measures that will keep risks below acceptable levels. Our RSP identifies critical capabilities, called Capability Thresholds, which tell us when we need to upgrade the protections for a given model. For each Capability Threshold, the RSP also specifies an overall safety target and particular categories of mitigations. This report focuses on the protections known as the ASL-3 Security and Deployment Standards (see Appendices A and B). These standards aim to prevent model misuse by making them harder to use for catastrophic harm and by protecting model weights from theft.

1.2 Decision to implement the ASL-3 Standard

Claude 3.7 Sonnet was sufficiently far below the "CBRN-3" capability threshold to be deployed under the ASL-2 Standard.[1] However, we noted that the model showed some improvement on biological weaponization proxy tasks and that progress on model capabilities suggested that future models would soon be near or cross the threshold. Following that launch, we decided that we would preemptively implement enhanced protections for our next, most advanced model, even if we had not yet determined that they were necessary. We did this for two reasons: First, we wanted to be prepared to apply these protections before they might actually be required. Second, we expect to iterate on and refine our model protections, and we wanted to jumpstart that process.

[1] The CBRN-3 Capability Threshold will be reached by models with "the ability to significantly assist individuals or groups with basic STEM backgrounds in obtaining, producing, or deploying CBRN weapons."

We are now activating the ASL-3 standard as part of launching Claude Opus 4. Claude Opus 4 showed clearly superior performance on some proxy CBRN tasks compared to Claude Sonnet 3.7. Moreover, some of our external red-teaming partners reported that Claude Opus 4 performed qualitatively differently from any model they previously tested.[2]

But dangerous capability evaluations of AI models are inherently challenging, and as models approach our thresholds of concern, it takes longer to determine their status. We are still evaluating whether the ASL-3 Standard is truly necessary for Claude Opus 4 or, instead, whether our preexisting baseline protections would suffice (in which case we may resume operating under ASL-2 protections). As required by the RSP for any model released under the ASL-3 standard, we have determined that Claude Opus 4 does not require the even-more-stringent protections of the ASL-4 Standard.

Section 2 provides additional detail about our implementation of the ASL-3 Deployment Standard. Section 3 does the same for the ASL-3 Security Standard. Section 4 concludes with a discussion of the rationale behind publishing this report.

2 Implementing the ASL-3 Deployment Standard

The RSP's ASL-3 Deployment Standard requires that models have safeguards making them "robust to persistent attempts to misuse the capability in question." Below, we describe the mitigations we've implemented to meet this standard, with particular attention to the risk of universal jailbreaks and our plan for ongoing monitoring and improvement of our deployment measures.

2.1 The CBRN capability threshold and the focus on universal jailbreaks

AI is a general purpose, dual-use technology. As noted in the International AI Safety Report 2025, “AI advances [are] beneficial to science while also lowering some barriers to chemical

[2] Our RSP evaluations indicated a gap in capabilities between Claude Opus 4 and Claude Sonnet 4, particularly in areas relevant to our RSP commitments. Whereas both models demonstrated improvements over Claude Sonnet 3.7, Claude Opus 4 showed substantially greater capabilities in CBRN-related evaluations, including stronger performance on virus acquisition tasks, more concerning behavior in expert red-teaming sessions, and enhanced tool use and agentic workflows. In contrast, Claude Sonnet 4 showed more modest improvements that—while noteworthy—did not approach the ASL-3 thresholds of concern.

and biological weapons development."[3] Addressing this risk is one of the key goals of the RSP.

The CBRN capability threshold for the ASL-3 Standard focuses on individuals or groups with basic technical backgrounds (e.g. undergraduate STEM degrees) attempting to use AI models to significantly help them create/obtain and deploy CBRN weapons. To avoid inadvertently increasing the likelihood of attempted misuse, we are not providing specific details on the threat models we have prioritized.[4] For purposes of this report, it suffices to note that the processes needed to generate these threats are knowledge-intensive, skill-intensive, prone to failure, and frequently have one or more bottleneck steps, judging from our consultation with people with extensive experience in biosecurity from Deloitte, SecureBio, and elsewhere, including in a meeting organized by the Frontier Model Forum with a broader set of experts. These *expert engagements*, which have been key to meeting the ***threat modeling*** aspect of the ASL-3 Deployment Standard, underscored that the bulk of the ways in which AI models could contribute to CBRN risk is by providing assistance with processes that take place on long time scales and require repeated interactions. This means that our ASL-3 deployment measures are not intended to prevent the extraction of commonly available single pieces of information, such as the answer to, "What is the chemical formula for sarin?" (although they often do prevent this).

For threat actors to obtain the requisite information from AI models, they would typically need to use some technique—referred to as a jailbreak or jailbreak technique—to bypass the protections that would otherwise prevent the model from sharing harmful information. Such jailbreaks are prompting techniques that essentially persuade the model to disregard its default constraints.

Accordingly, to defend against ASL-3 threats, we have implemented additional protections that are much more challenging to jailbreak. These protections guard against *universal jailbreaks* in particular. Universal jailbreaks are systematic prompting strategies that can reliably bypass an AI system's safety mechanisms (including both the model and surrounding classifiers).

[3] Bengio, Y., et al. (2025). International AI Safety Report. Department for Science Innovation and Technology Report No. DSIT 2025/001. https://www.gov.uk/government/publications/international-ai-safety-report-2025, p. 25.
[4] We will continue to assess if there are significant changes related to some of our threat modeling uncertainties, for example, related to the length of model access required to achieve uplift, the number of potential threat actors, and changes in the complexity of the threat pathway. Ongoing consultations with experts and monitoring of traffic on our platforms will inform these judgments.

Furthermore, our primary intention is to limit the availability of *highly-effective* universal jailbreaks. Our red-teaming has shown that discovering universal jailbreaks requires significant effort from experts, and that some jailbreaks significantly degrade model capabilities. But jailbreaks that result in substantial reductions in model capabilities are unlikely to uplift a threat actor. We have implemented a bug bounty program to incentivize the responsible discovery and reporting of highly-effective universal jailbreaks.

2.2 Deployment measures

To protect models under the ASL-3 Deployment Standard, we have taken a three-part approach: making the discovery of universal jailbreaks relevant to our threat models much harder, identifying jailbreaks when they are found, and using that information to improve our defenses. Details about these mitigations and processes follow. Combined with the preexisting protections of the ASL-2 Deployment Standard (see Appendix C), these measures form a ***defense in depth*** against catastrophic misuse of the model, as is called for in the ASL-3 Deployment Standard (see Appendix A).

The remainder of this section provides further information about these mitigations.

2.2.1 Real-time classifier guards

2.2.1.1 How they work

Real-time classifier guards based on Constitutional Classifiers are deployed for Claude Opus 4.[5] These are large language models that monitor model inputs and outputs in real time and block the model from producing a narrow range of harmful information relevant to our threat model.[6] Constitutional Classifiers are central to our ASL-3 deployment protections, as we believe they will make universal jailbreaks that do not degrade model capabilities substantially harder to develop.

To develop these classifiers, we began by identifying three types of uses of our models: (1) prohibited uses; (2) uses that could be permissible but are also relevant to our threat models; and (3) clearly permissible uses. These comprise what we call the constitution. We then used this constitution to generate synthetic examples of prompts and completions in each of the three categories. Finally, we trained the Constitutional Classifiers, which are LLMs fine tuned to block harmful interactions related to our threat model while allowing permitted ones.[7]

[5] Sharma, M., et al. (2025). Constitutional Classifiers: Defending against Universal Jailbreaks across Thousands of Hours of Red Teaming. arXiv:2501.18837. https://arxiv.org/abs/2501.18837

[6] Initially they are focused exclusively on biological weapons, although we are evaluating a potential expansion in scope to some other CBRN threats.

[7] For more details on how to implement these classifiers, see Sharma, et al. (2025), cited above.

We conducted extensive *red-teaming during research and development* of our Constitutional Classifier-based approach. This is part of how we addressed the ASL-3 Deployment Standard requirement for ***red-teaming*** of our deployment measures, along with pre-deployment testing of updated real-time classifiers and the bug bounty that we launched to encourage further red-teaming. This research showed that our classifiers make it harder to find jailbreaks, but no classifier is perfect. For example, our bug bounty and other red-teaming have revealed a small number of potentially effective jailbreaks. We have identified effective remediations for the vulnerabilities revealed this way. See the Deployment Safeguards Report for more details.

2.2.1.2 Implementation

As in any classifier system, false positives (instances where legitimate content is incorrectly flagged) are inevitable. We have reduced false positives significantly from the research to the production phase and expect them to only affect a narrow range of traffic. We will gather user feedback on false positives and continually refine our classifiers to protect against potential harm while minimizing disruption to legitimate uses.

Classifiers add compute overhead (additional processing costs beyond what is required for model inference) to the system, but our pre-production testing suggests we can substantially reduce jailbreaking success while adding only moderate compute overhead to normal operations.

2.2.1.3 Future directions

Real-time classifier guards are a first step. Building on this work, we have already started to implement approaches for improving the classifiers: automated red-teaming, better policy training, expanded synthetic data training, upgrading the classifiers to more capable models, and getting feedback from customers.

2.2.2 Ongoing assessment

We are adopting a multi-pronged approach to assessing and improving our real-time classifier guards on an ongoing basis as part of the ASL-3 Deployment Standard requirement for ***monitoring*** of our system's performance.

2.2.2.1 Offline monitoring

Offline monitoring systems complement our real-time classifiers. Our monitoring tracks harmful interactions, subject to retention and privacy constraints. Because offline monitors are not subject to the same compute and latency constraints as our real-time systems, they are able to achieve improved performance, though they do not enable us to intervene right away in specific cases.

Offline monitoring enables us to assess the frequency of real-time classifier safeguards breaches, and in some cases to improve classifiers by identifying jailbreaks. They serve as an important backstop and part of our overall defense-in-depth approach. Monitoring also allows us to identify the rate of false positives, which will enable us to improve false positive rates over time.

2.2.2.2 Bug bounty program

Following best practices for securing high-value software, we have implemented a *bug bounty program* with substantial rewards for reporting universal jailbreaks of concern that evade our real-time defenses. The bug bounty program uses a set of test queries related to our threat model. We partnered with HackerOne to run a pre-launch version of this bug bounty using our ASL-3 real-time classifiers guarding Claude Sonnet 3.7. This program will now convert to an ongoing bug bounty testing our deployed classifiers on Claude Opus 4. We began this new bug bounty initiative with participation from the researchers who joined our earlier program last year and are offering the opportunity for new researchers to participate. If you are an experienced red teamer or have demonstrated expertise in identifying jailbreaks in language models, we encourage you to apply for an invitation through our application form.

The bug bounty program incentivizes several useful behaviors. First, it encourages the responsible discovery and reporting of novel jailbreaks and publicly available jailbreaks, including jailbreaks that might otherwise have been unreported, sold, or otherwise shared. Further, this program will incentivize red-teamers to continuously stress test our real-time classifier systems, providing a regular signal for how well (or poorly) our classifiers are working.

2.2.2.3 Threat intelligence

We are working with *threat intelligence* vendors to monitor public forums and black markets for publicly available model jailbreaks, particularly universal jailbreaks that target our key threat models.

2.2.3 Rapid response

When jailbreaks are discovered, we will triage them based on severity and apply *jailbreak rapid response techniques*,[8] aiming to reinforce our classifiers against them within days to

[8] Peng, A., et al. (2024). Rapid Response: Mitigating LLM Jailbreaks with a Few Examples. arXiv:2411.07494. https://arxiv.org/abs/2411.07494

weeks depending on the assessed threat level and other factors. One rapid response approach is "jailbreak proliferation," in which an LLM generates large numbers of additional examples similar to observed attacks. We use these synthetically generated jailbreaks to train a classifier model that blocks similar inputs. Our research shows that this substantially reduces attack success for in-distribution jailbreaks and also (to a lesser degree) for out-of-distribution jailbreaks. This research is an important part of why we believe ***rapid remediation*** is feasible, as required by the ASL-3 Deployment Standard. Our new bug bounty program will also provide data against which to evaluate our ability to rapidly patch jailbreaks as they are discovered.

2.2.4 Access controls

Dual-use science and technology applications sometimes require access to model capabilities that overlap with potentially harmful uses. Rather than implementing blanket restrictions that might hinder legitimate use, we have developed a verification system to enable controlled access where appropriate. This system offers controlled access to entitles with which we have an established relationship and that have gone through a process of verifying organization legitimacy, basic security measures, and a history of compliance with our usage policy.

Once vetted, qualified users can receive targeted exemptions from certain classifier actions, with ongoing monitoring for compliance with usage policies. This approach to *access controls* meets the ASL-3 Deployment Standard goal of establishing criteria for determining how to provide differentiated access to ***trusted users***.

We reserve the right to turn on real-time classifier guards should we identify potential misuse or violations of the exemption scope.

2.3 Discussion

We believe that "the measures we have implemented make us robust to persistent attempts to misuse" the model's CBRN-related capabilities, as required under the ASL-3 Deployment Standard (see Appendix A). Table 1 shows how our deployment-related work maps on to the specific criteria of the ASL-3 Deployment Standard. See the ASL-3 Deployment Safeguards Report for a more complete discussion of the sufficiency of these measures.

3 Implementing the ASL-3 Security Standard

Under the RSP's ASL-3 Security Standard, we must implement measures that make us highly protected against most attackers' attempts at stealing model weights. In this section,

ASL-3 Deployment Standard Criterion	**Relevant Measures**
Threat modeling	● Expert engagements
Defense in depth	● Real-time classifier guards ● Offline monitoring ● Bug bounty program ● Threat intelligence
Red-teaming	● Red-teaming during research and development ● Pre-deployment red-teaming ● Bug bounty program
Rapid remediation	● Jailbreak rapid response techniques
Monitoring	● Offline monitoring ● Bug bounty program ● Threat intelligence
Trusted users	● Access controls
Third-party environments	● Real-time classifier guards consistent across environments ● Uniform access controls ● Bug bounty program and threat intelligence are not environment-specific ● Offline monitoring and jailbreak rapid response on all environments with some technical implementation differences

Table 1 Deployment measures mapped to criteria in the ASL-3 Deployment Standard

we explain our threat model and priorities, provide an overview of our protections, and discuss our future work. As with our deployment measures, we recognize that implementing the ASL-3 Security Standard, while a good first step, is only the beginning of our efforts to secure our models from relevant threats. We plan to expand our efforts and look forward to continuing this work.

3.1 Threat modeling and risk assessment

We began with threat modeling and risk assessment to identify the security safeguards necessary to appropriately protect ASL-3 models. This section summarizes the work done to meet the ***threat modeling*** criterion in the ASL-3 Security Standard (Appendix B).

3.1.1 Threat actors

The RSP identifies specific threat actors considered in-scope for ASL-3, including hacktivists, criminal hacker groups, organized cybercrime groups, terrorist organizations, corporate espionage teams, basic insiders, and undifferentiated attacks from state-sponsored groups. Sophisticated insiders, state-compromised insiders, nation-state attackers, and advanced persistent threat (APT)-level actors are considered out of scope for ASL-3.[9]

3.1.2 Asset identification

The ASL-3 Security Standard focuses on model weights—the trained numerical parameters that embody our AI's intelligence and capabilities. If stolen, model weights could enable malicious actors to bypass our monitoring and safeguards by deploying the model on external infrastructure. This, in turn, would enable the kind of persistent and unfettered access that would allow for the multi-turn, harmful interactions central to our threat model.

Although core to the ASL-3 Standard, and therefore this report, protection of model weights is only one aspect of Anthropic's security program. Our broader security approach encompasses cloud infrastructure, protection of non-model-weight IP, customer data and other priorities.

3.1.3 Attack vectors

We considered six primary attack vectors. Together, these cover attack phases of initial compromise (1-4), lateral movement or permission elevation (5), and target acquisition and exfiltration (6).

1. **Compromise of employee devices**: This vector focuses on attacks targeting laptops, phones, and other endpoint devices used by employees. Specific techniques include malware infection through compromised files, credential theft via phishing, and physical device theft, among others.
2. **Supply chain attacks**: These target upstream dependencies or build systems that Anthropic relies on, including compromising upstream dependencies (like software libraries), trojanizing development tools, and attacking cloud infrastructure providers.

[9] For further discussion of the distinction between basic and sophisticated insiders, see Anthropic's Responsible Scaling Policy, Version 2.2.

3. **Physical attacks**: This vector includes breaching office locations to steal devices or place surveillance implants, and coercing employees to assist in attacks.
4. **Compromise of cloud-hosted infrastructure and services**: This covers attacks against internet-facing servers, APIs used in model hosting or internal tools, and misconfigured cloud resources, among others.
5. **Privilege escalation**: The risk here is that attackers with legitimate but limited access can expand their reach to access sensitive data like model weights.
6. **Exfiltration of sensitive data**: The final vector addresses how attackers might steal intellectual property through network-based exfiltration, compromised endpoints, or exploitation of third-party services.

Internal development of these threat models using industry standard practices are followed by *external partner consultation* to validate comprehensive coverage. We also used *red team exercises* to simulate attacks and discover control oversights. *Continuous evaluation and regular updates* of the threat models improve the accuracy with which they characterize practical threats.

3.2 Selected security measures

In this section, we describe some of the key updates to our security program that we've made as part of our implementation of the ASL-3 Security Standard. These measures, part of a broader set of more than 100 new controls, are designed to reduce risk from the attack vectors above. As with any security program, our efforts will evolve in both design and implementation, and we will continue refining and strengthening our protections.

These measures build on the large number of controls and practices that we had already implemented as part of the ASL-2 Security Standard (read more at trust.anthropic.com).

Here are some specific key enhancements we have implemented as part of the ASL-3 Standard. Many of these controls are examples of following the best practices established by other security-conscious organizations, although we believe the first is a more unique, experimental control.

- **Egress bandwidth controls.** We have implemented preliminary egress bandwidth controls, which we will continue to improve. Egress bandwidth controls restrict the flow of data out of secure computing environments where AI model weights reside. There are several advantages to this:
  - Implementing egress bandwidth controls has been a forcing function to understand and control the way in which data is flowing outside of our internal systems, which has yielded benefits for our detection and response capabilities.

    - Over time, we expect to get to the point where rate limits are low enough that exfiltrating model weights before being detected is very difficult - even if an attacker has compromised our systems significantly otherwise. The combined weights of a model are substantial in size. By limiting the rate of outbound network traffic, these controls can leverage model weight size to create a security advantage. When potential exfiltration is detected through unusual bandwidth usage, security systems can block the suspicious traffic, addressing the data exfiltration attack vector.
    - Eventually, it may be necessary to restrict model weights and other key assets to airgapped networks. Egress bandwidth controls are a step in this direction and are helping to derisk and prepare for such a possibility.
- **Two-party control.** This system automates the requirement that any employee needing access to model weights must authenticate with a physical security key, provide a justification, and obtain authorization from a second party at the time of the request.[10] Permission, when granted, is given for only a limited period, timing out automatically. This augments detection and response signals by reducing the attack monitoring surface and adds peer review to approve high-privilege access, making a compromise of any one user account less likely to result in exposure to model weights.
- **Endpoint software control.** We have implemented binary allowlisting mechanisms. This deployment, along with our endpoint detection and response (EDR) software, blocks execution of less sophisticated malware on Anthropic devices. Industry experience has shown that these controls are particularly effective at mitigating the risk of unauthorized or malicious software and they are a key part of our security strategy. This control helps guard against the employee device compromise attack vector.
- **Change management for secure development.** For code repositories containing model weights and security infrastructure, we have implemented several changes to facilitate appropriate review of code changes. These protections include requiring additional reviews on some changes, mandatory cryptographically signed commits, and designated team ownership for critical paths. To maintain productivity, we are using Claude to augment our review of routine code changes, especially for less sensitive files, improving the efficiency of re-review of minor updates to previously approved pull requests. These measures add to our defense against the supply chain attack threat vector without significantly reducing engineering velocity.

[10] See our July 2023 post "Frontier Model Security," for further discussion of this control.

These measures are designed to reduce risk from the attack vectors listed in the previous section. As with any security program, our efforts will evolve in both design and implementation, and we will continue refining and strengthening our protections.

Below, we outline some of the additional measures we have implemented, organizing them in accordance with the ***Security Frameworks*** identified in the ASL-3 Security Standard. For reasons of space and because these approaches are familiar to security professionals, we summarize these measures at a high level.

3.2.1 Perimeters and access controls

The ASL-3 Security Standard includes ***perimeter*** and ***access controls*** to protect model weights and critical systems from unauthorized access. As noted above, many of these controls follow established best practices. These include *network security* measures such as segmentation, encryption, and network storage and hardening. Configuration standards, restricted storage locations, and security review of infrastructure changes contribute to *cloud security*. *Physical security* controls include layered office security, monitoring, and guest management, along with procedures for secure destruction of media. Other *digital access controls* include role-based access with least privilege, multi-factor authentication with hardware keys, session timeout enforcement, single sign-on integration, and context-aware access controls. As discussed above, our use of *two-party control* is especially relevant for the ASL-3 Standard's emphasis on protecting model weights.

3.2.2 Lifecycle security

The ASL-3 Security Standard also includes ***lifecycle security***, emphasizing the need to secure the links in the chain of systems and software used to develop AI models. Some of the *hardware security* measures we have used are procurement from a restricted list of vetted vendors and managed laptop configurations. *Endpoint security* controls include binary allowlisting (see above), device health monitoring, endpoint detection and response, and device authorization controls. Dependency scanning, package verification, and secure development standards are aspects of how we have approached *supply chain security*. A secure build pipeline with signature verification (using the SLSA framework) is important for *build pipeline security*. Another area of emphasis for lifecycle security has been *change management for secure development*, discussed above.

3.2.3 Monitoring

Proactively identifying and mitigating threats requires ***monitoring***, another framework noted by the ASL-3 Security Standard. Many of our monitoring measures follow standard practices. These measures include *centralized logging of security events* with automated analysis, creation of *automated alerts* for anomalous and suspicious activity, and the use of *intrusion detection systems* for continuous network monitoring. *Monitoring access to critical assets* including model weights is a key task; our preliminary implementation of egress bandwidth controls and continued use of two-party control make this task more manageable. Finally, *deception technology* such as honeypots aid in proactive identification of threats.

3.2.4 Resourcing

The ASL-3 Security Standard anticipated that 5-10% of employees would need to be dedicated to security. As of this writing, 5.2% of Anthropic employees work on security and security-adjacent roles, fulfilling this commitment to ***resourcing***.

3.2.5 Existing guidance

As noted previously, many of the measures discussed above are part of standard security practices at other companies and we have been able to draw on the standards and ***existing guidance*** already used in the industry, as called for by the ASL-3 Security Standard. Anthropic has achieved several *compliance attestations and certifications*, including the SOC 2 Type 2, CSA STAR Level 2, ISO 27001, and ISO 42001 certifications, The security program and supporting controls also incorporate standards from NIST 800-53, the Secure Software Development Framework (SSDF), and other recognized cybersecurity frameworks, contributing to *industry framework alignment*.

3.2.6 Audits

The ASL-3 Security Standard requires plans for ***audits*** of the design and implementation of the security program. Anthropic has already benefitted from audits and other engagements with third party experts. Previous third-party audits of our initial ASL-3 design of controls revealed important suggestions for improving the design of our ASL-3 controls. Some of the certifications referenced in the previous section required audits (notably SOC 2 Type 2, CSA STAR Level 2, ISO 27001, and ISO 42001).

As is required by best practice and per our ISO 27001 and ISO 42001 certifications, we will conduct ongoing risk assessments at a frequency commensurate with the rate of change of relevant threats and attack vectors. We will continue to engage third parties for external review of our risk assessments, the resulting updates to the design of our security measures, and their effectiveness. This *external validation and auditing* feeds into a process of *continuous improvement*. The security program has a defined process to triage, investigate, and remediate audit findings from periodic audits. Anthropic also incorporates

ASL-3 Security Standard Criterion	Relevant Measures
Threat modeling	● Threat actor identification ● Asset identification ● Attack vector identification ● External partner consultation ● Red team exercises ● Continuous evaluation and regular updates
Security frameworks	*Perimeters and access controls* ● Digital access controls ● Network security ● Cloud security ● Physical security *Lifecycle security* ● Hardware security ● Endpoint security ● Supply chain security ● Build pipeline security ● Change management for secure development *Monitoring* ● Centralized logging of security events ● Automated alerts ● Intrusion detection systems ● Monitoring access to critical assets ● Deception technology *Resourcing* ● Security is staffed at 5.2% of total employees *Existing guidance* ● Compliance attestations and certifications ● Industry framework alignment
Audits	● External validation and auditing ● Continuous improvement ● Penetration testing ● Bug bounty
Third-party environments	● Third-party partners must implement equivalent security controls including multi-factor authentication, multi-party

	code review, role-based access, integrity verification, logical separation, insider threat protections, audit logging, penetration testing, and access approval controls for model deployments.

Table 2 Security measures mapped to criteria in the ASL-3 Security Standard

recommendations from external security experts to maintain alignment with evolving industry guidance.

In particular, we will continue regularly engaging with world-class penetration testers and security red teamers through our *penetration testing* and *bug bounty* programs. These experts probe our infrastructure and product for weaknesses that could compromise our model weight protections. They perform simulated attacks against Anthropic to validate the effective implementation of our security safeguards or find overlooked weaknesses.

3.3 Discussion

As with any security program, our efforts will evolve in both design and implementation, and we will continue refining and strengthening our protections. Beyond some of the improvements to the initial round of ASL-3 controls discussed above, we are continuing to mature our security program with steps including:

- resolving and reducing exceptions and bypasses to our controls,
- improving visibility of metrics ensuring the timely resolution of exceptions,
- reducing risk introduction-discovery-treatment cycle times,
- increasing the use of automation for both the implementation and monitoring of safeguards compliance, and
- increasing the frequency of engagement with external experts for tasks including risk assessment, auditing, and penetration testing.

Table 2 summarizes how selected security measures map to the criteria in the ASL-3 Security Standard (see Appendix B). This table includes measures discussed in Section 3 as well as others, but it does not include all controls put in place to prepare for ASL-3.

4 Conclusion

Implementing the ASL-3 Standard is an important milestone. As we move forward, we will continue to refine our ASL-3 protections while deepening our understanding of even more advanced models and the safeguards they might warrant.

In publishing this report, moreover, we are not declaring that we have achieved total assurance against the threat models of concern: there will always be new risks to consider and improvements to be made. Nonetheless, we are providing this update on our current progress to advance the societal dialogue on the safe development of frontier AI systems. Our collective ability to achieve this mission will require a healthy discourse between frontier model providers, policymakers, civil society, and the public at large. This report complements our other efforts to share our learnings and experiences with others, including through the Frontier Model Forum.

We look forward to feedback on our approach as well as to learning from the approaches and experiences of other companies tackling similar problems.

Appendices

Appendix A: ASL-3 Deployment Standard

(Quoted in full from Anthropic's RSP, Version 2.2)

When a model must meet the ASL-3 Deployment Standard, we will evaluate whether the measures we have implemented make us robust to persistent attempts to misuse the capability in question. To make the required showing, we will need to satisfy the following criteria:

1. **Threat modeling:** Make a compelling case that the set of threats and the vectors through which an adversary could catastrophically misuse the deployed system have been sufficiently mapped out, and will commit to revising as necessary over time.
2. **Defense in depth:** Use a "defense in depth" approach by building a series of defensive layers, each designed to catch misuse attempts that might pass through previous barriers. As an example, this might entail achieving a high overall recall rate using harm refusal techniques. This is an area of active research, and new technologies may be added when ready.
3. **Red-teaming:** Conduct red-teaming that demonstrates that threat actors with realistic access levels and resources are highly unlikely to be able to consistently elicit information from any generally accessible systems that greatly increases their ability to cause catastrophic harm relative to other available tools.[11]
4. **Rapid remediation:** Show that any compromises of the deployed system, such as jailbreaks or other attack pathways, will be identified and remediated promptly enough to prevent the overall system from meaningfully increasing an adversary's ability to cause catastrophic harm. Example techniques could include rapid vulnerability patching, the ability to escalate to law enforcement when appropriate, and any necessary retention of logs for these activities.
5. **Monitoring:** Prespecify empirical evidence that would show the system is operating within the accepted risk range and define a process for reviewing the system's performance on a reasonable cadence. Process examples include monitoring responses to jailbreak bounties, doing historical analysis or background monitoring, and any necessary retention of logs for these activities.

[11] This criterion does not attempt to specify the exact red-teaming protocol (e.g., number of hours, level of access, or pass-fail criteria). Setting a principled pass-fail threshold will depend on other factors, such as the quality of our monitoring and ability to respond to jailbreaks rapidly. Due to the likely ease of bypassing or removing safeguards via fine-tuning, it may be difficult or impossible for these red-teaming tests to pass if weights are released or if unmoderated fine-tuning access is provided to untrusted users.

6. **Trusted users:** Establish criteria for determining when it may be appropriate to share a version of the model with reduced safeguards with trusted users. In addition, demonstrate that an alternative set of controls will provide equivalent levels of assurance. This could include a sufficient combination of user vetting, secure access controls, monitoring, log retention, and incident response protocols.
7. **Third-party environments:** Document how all relevant models will meet the criteria above, even if they are deployed in a third-party partner's environment that may have a different set of safeguards.

Appendix B: ASL-3 Security Standard

(Quoted in full from Anthropic's RSP, Version 2.2)

When a model must meet the ASL-3 Security Standard, we will evaluate whether the measures we have implemented make us highly protected against most attackers' attempts at stealing model weights.

We consider the following groups in scope: hacktivists, criminal hacker groups, organized cybercrime groups, terrorist organizations, corporate espionage teams, internal employees,[12] and state-sponsored programs that use broad-based and non-targeted techniques (i.e., not novel attack chains).

The following groups are out of scope for the ASL-3 Security Standard because further testing (as discussed below) should confirm that the model would not meaningfully increase their ability to do harm: state-sponsored programs that specifically target us (e.g., through novel attack chains or insider compromise) and a small number (~10) of non-state actors with state-level resourcing or backing that are capable of developing novel attack chains that utilize 0-day attacks.

To make the required showing, we will need to satisfy the following criteria:

1. **Threat modeling:** Follow risk governance best practices, such as use of the MITRE ATT&CK Framework to establish the relationship between the identified threats, sensitive assets, attack vectors and, in doing so, sufficiently capture the resulting risks that must be addressed to protect model weights from theft attempts. As part of this requirement, we should specify our plans for revising the resulting threat model over time.
2. **Security frameworks:** Align to and, as needed, extend industry-standard security frameworks for addressing identified risks, such as disclosure of sensitive information, tampering with accounts and assets, and unauthorized elevation of privileges with the appropriate controls. This includes:
    a. **Perimeters and access controls:** Building strong perimeters and access controls around sensitive assets to ensure AI models and critical systems are

[12] We will implement robust controls to mitigate basic insider risk, but consider mitigating risks from sophisticated or state-compromised insiders to be out of scope for ASL-3. We define "basic insider risk" as risk from an insider who does not have persistent or time-limited access to systems that process model weights. We define "sophisticated insider risk" as risk from an insider who has persistent access or can request time-limited access to systems that process model weights. We are committed to further enhancing these protections as a part of our ongoing preparations for higher security levels.

protected from unauthorized access. We expect this will include a combination of physical security, encryption, cloud security, infrastructure policy, access management, and weight access minimization and monitoring.

   b. **Lifecycle security:** Securing links in the chain of systems and software used to develop models, to prevent compromised components from being introduced and to ensure only trusted code and hardware is used. We expect this will include a combination of software inventory, supply chain security, artifact integrity, binary allowlisting, hardware procurement, and secure research development lifecycle.
   c. **Monitoring:** Proactively identifying and mitigating threats through ongoing and effective monitoring, testing for vulnerabilities, and laying traps for potential attackers. We expect this will include a combination of endpoint patching, product security testing, log management, asset monitoring, and intruder deception techniques.
   d. **Resourcing:** Investing sufficient resources in security. We expect meeting this standard of security to require roughly 5-10% of employees being dedicated to security and security-adjacent work.
   e. **Existing guidance:** Aligning where appropriate with existing guidance on securing model weights, including Securing AI Model Weights, Preventing Theft and Misuse of Frontier Models (2024); security recommendations like Deploying AI Systems Securely (CISA/NSA/FBI/ASD/CCCS/GCSB /GCHQ), ISO 42001, CSA's AI Safety Initiative, and CoSAI; and standards frameworks like SSDF, SOC 2, NIST 800-53.

3. **Audits:** Develop plans to (1) audit and assess the design and implementation of the security program and (2) share these findings (and updates on any remediation efforts) with management on an appropriate cadence. We expect this to include independent validation of threat modeling and risk assessment results; a sampling-based audit of the operating effectiveness of the defined controls; periodic, broadly scoped, and independent testing with expert red-teamers who are industry-renowned and have been recognized in competitive challenges.
4. **Third-party environments:** Document how all relevant models will meet the criteria above, even if they are deployed in a third-party partner's environment that may have a different set of safeguards.

Appendix C: ASL-2 Deployment Standard

(Quoted in full from Anthropic's RSP, Version 2.2)

1. **Acceptable use policies and model cards:** Publication of model cards for significant new models describing capabilities, limitations, evaluations, and intended use cases. Enforcement of a Usage Policy that restricts, at a minimum, catastrophic and high harm use cases, including using the model to generate content that could cause severe risks to the continued existence of humankind, or direct and severe harm to individuals.
2. **Harmlessness training and automated detection:** Training models to refuse requests to aid in causing harm, such as with Constitutional AI or other improved techniques, and the use of model enhanced trust and safety detection and enforcement.
3. **Fine-tuning protections:** In finetuning products, data is filtered for harmfulness, and models are subject to automated evaluation to check harmlessness features are not degraded. There are a very limited number of use cases where this tooling is disabled. These are negotiated on a case by case basis and considered only for extremely low risk use cases that involve company personnel.
4. **Vulnerability reporting channels:** Clearly indicated paths within the product for users to report harmful or dangerous model outputs, as well as a bug bounty for universal jailbreaks.

Appendix D: ASL-2 Security Standard

(Quoted in full from Anthropic's RSP, Version 2.2)

A security system that can likely thwart most opportunistic attackers.

1. **Supply chain:** Vendor and supplier security must be regularly reviewed to ensure that they meet security standards. Software updates should be frequently managed and compliance monitoring automated where possible.
2. **Offices:** Physical security should entail visitor access logs and restrictions protect on-site assets. Highly sensitive interactions should utilize advanced authentication like security keys. Network visibility should be maintained and office access controls and communications should maximize on-site protections.
3. **Workforce:** People-critical processes must represent a key aspect of cybersecurity. Mandatory periodic infosec training educates all employees on secure practices, like proper system configurations and strong passwords, and fosters a proactive "security mindset." Fundamental infrastructure and policies promoting secure-by-design and secure-by-default principles should be incorporated into the engineering process. An insider risk program should tie access to job roles. Rapid incident response protocols
4. **Compartmentalization:** Segmented system isolation must ensure limited blast radius. Features like zero trust architecture should require access from approved devices. Strict protocols must be deployed to regulate weight copies on company networks and limit storage to only approved, restricted systems.
5. **Infrastructure:** Standard security infrastructure, monitoring software, access management tools, and disk encryption provide a technology baseline. Process elements like incident reporting procedures, lost/stolen device protocols and Detection and Response should support these. External validation like SOC 2 compliance and continuous vulnerability management must ensure adaptations match infosec developments. Programs like bug bounties and vulnerability discovery should incentivize exposing flaws.
6. **Operations:** Ongoing configuration management, compliance drills, integrated security approaches and mandatory external reviews should embed security within regular operations and harden processes during organizational changes.